MORD: Multi-class classifier for Ordinal regression

Kostiantyn Antoniuk, Vojtěch Franc, and Václav Hlaváč

Czech Technical University in Prague,
Faculty of Electrical Engineering,
Technicka 2, 166 27, Praha 6
{antonkos,xfrancv,hlavac}@cmp.felk.cvut.cz

Abstract. We show that classification rules used in ordinal regression are equivalent to a certain class of linear multi-class classifiers. This observation not only allows to design new learning algorithms for ordinal regression using existing methods for multi-class classification but it also allows to derive new models for ordinal regression. For example, one can convert learning of ordinal classifier with (almost) arbitrary loss function to a convex unconstrained risk minimization problem for which many efficient solvers exist. The established equivalence also allows to increase discriminative power of the ordinal classifier without need to use kernels by introducing a piece-wise ordinal classifier. We demonstrate advantages of the proposed models on standard benchmarks as well as in solving a real-life problem. In particular, we show that the proposed piece-wise ordinal classifier applied to visual age estimation outperforms other standard prediction models.

Keywords: Ordinal regression, linear multi-class classification.

1 Introduction

The classification problem consists of predicting a hidden class label $y \in \mathcal{Y}$ based on observations $\boldsymbol{x} \in \mathcal{X}$ using a classifier $h\colon \mathcal{X} \to \mathcal{Y}$. In the statistical classification, the pairs of $(\boldsymbol{x}, y)$ are assumed to be a realization of some random variables distributed according to $P(\boldsymbol{x}, y)$. This paper analyses a class of classification problems fitting under the ordinal regression setting which imposes additional assumptions on the distribution $P(\boldsymbol{x}, y)$. In particular, the labels in $\mathcal{Y}$ are assumed to be ordered, w.l.o.g. we use $\mathcal{Y} = \{1, \ldots, Y\}$ equipped with a natural order, and they are modeled as a result of a course measurement of some continuous random variable $\chi(\boldsymbol{x})$. More precisely, let as define a set of Y intervals

$$U(1) = (-\infty, \theta_1]\,, \;\; U(2) = (\theta_1, \theta_2], \;\; \ldots, \;\; U(Y) = (\theta_{Y-1}, \infty)\,,$$

determined by a sequence of non-decreasing thresholds $\theta_1,\ \theta_2, \ldots, \theta_{Y-1}$. The standard model of [1] assumes that we observe label $y \in \mathcal{Y}$ if a realization of the random variable $\chi(\boldsymbol{x})$ is in the interval $U(y)$. Thus the classes correspond to

contiguous ordered intervals on some continuous scale. Based on this assumption various ordinal regression models have been proposed and they are routinely applied in fields like social sciences, epidemiology, information retrieval or, recently in computer vision.

A typical problem is how to learn the classifier given a set of training examples $\{(\boldsymbol{x}^1, y^1), \ldots, (\boldsymbol{x}^m, y^m)\} \in (\mathcal{X} \times \mathcal{Y})^m$ drawn from i.i.d. random variables distributed according to some unknown distribution $P(\boldsymbol{x}, y)$ which satisfies the "ordering" assumption mentioned above. In this paper, we consider the formulation which defines the target classifier to be the one with minimal expected risk (also called Bayes classifier)

$$R(h) = E_{p(\boldsymbol{x},y)}\big(\Delta(y, h(\boldsymbol{x}))\big)$$

where $\Delta\colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ is a given application specific loss function penalizing responses of the classifier.

In statistics, the learning problem is typically solved by constructing a plug-in Bayes classifier which replaces the true distribution $P(\boldsymbol{x}, y)$ by its Maximum-Likelihood estimate. This approach requires to guess the shape of the underlying distribution $P(\boldsymbol{x}, y)$ which can be difficult in practice. A different approach based on the risk minimization paradigm has been put forward in the machine learning literature. The idea is to learn the classifier directly from the examples without the need to estimate the generating distribution [2]. This approach selects the best classifier from a prescribed class of classifiers by minimizing a surrogate of the expected risk $R(h)$. The typical class of classifiers considered in the context of ordinal regression is the linear thresholded rule

$$h(\boldsymbol{x}; \boldsymbol{w}, \boldsymbol{\theta}) = 1 + \sum_{k=1}^{Y-1} [\![\langle \boldsymbol{x}, \boldsymbol{w} \rangle > \theta_k]\!] \,, \tag{1}$$

where $\boldsymbol{x} \in \mathcal{X} = \mathbb{R}^n$ is a vector of real-valued features, $\boldsymbol{w} \in \mathbb{R}^n$ is a parameter vector and $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_{Y-1}) \in \mathbb{R}^{Y-1}$ a vector of thresholds. In the sequel we refer to (1) as the ordinal (ORD) classifier. We call the vector $\boldsymbol{\theta}$ admissible iff its components are non-decreasing i.e. $\boldsymbol{\theta} \in \Theta = \{\boldsymbol{\theta}' \in \mathbb{R}^{Y-1} \mid \theta'_k \leq \theta'_{k+1}, k = 1, \ldots, Y-1\}$. The form of the ORD classifier reflects the assumption that the classes correspond to intervals on $\mathbb{R}$. It is seen that ORD classifier predicts y iff the value $\langle \boldsymbol{x}, \boldsymbol{w} \rangle$ is in the interval $U(y)$.

A Perceptron-like algorithm called PRank learning the ORD classifier in an on-line fashion has been proposed in [3]. They formulate learning as minimization of the empirical risk with the Mean Absolute Error (MAE) loss function $\Delta(y, y') = |y - y'|$ (also called ranking loss) and provide mistake bounds for the case of separable examples. The authors of [4] proposed to learn the ORD classifier by a modified Support Vector Machine algorithm originally designed for two-class classification. The paper [5] improves the algorithms of [4] by enforcing the learned thresholds to be admissible. A generic framework which allows to convert learning of the ORD classifier to the problem of learning a two-class linear SVM classifier (with modified example weights) have been proposed in [6].

They show that appropriately weighted SVM hinge-loss is an upper bound of so called V-shaped loss (e.g. MAE and the 0/1-loss are V-shaped) evaluated on the ORD classifier. The paper [7] analyses a relation between the ordinal regression and the multi-class classifiers, however, by their definition the ordinal classifier as any Bayesian classifier with the V-shaped loss function, i.e. they do not considered ordering of the labels at all.

The previous works in their core convert learning of the ORD classifier into learning a set of two-class classifiers. The resulting two-class classifiers are trained by a modified SVM algorithm [4][5][6] or Perceptron [3]. In this paper we show that such conversions are not necessary. We prove that the ORD classifier is equivalent to a linear multi-class classifier whose class parameter vectors are collinear and their magnitude is linearly increasing with the label. We call the new representation the Multi-class ORDinal (MORD) classifier. Our equivalence proof is constructive so that we can convert any ORD classifier to the MORD classifier and vice-versa. We show that the new representation can be beneficial for learning. In particular, the well understood methods for learning multi-class linear classifiers can be readily applied. We experimentally show that a generic multi-class SVM algorithm used to learn MORD delivers the same (or slightly better) results when compared to the specialized learning algorithms derived for the ORD classifier. The proposed approach works for (almost) arbitrary loss function unlike the existing methods which require V-shaped losses. In addition, we show that the new representation allows to increase discriminative power of the ordinal classifier without need to use kernels by introducing a piece-wise ordinal classifier. We demonstrate advantages of the proposed models on standard benchmarks as well as in solving a real-life problem. We show that the proposed piece-wise ordinal classifier applied to visual age estimation outperforms other prediction models and is also comparable to commercial solutions.

The paper is organized as follows. The equivalence between the ORD classifier and the linear multi-class classifiers is described in Section 2. In Section 3, we define a new model for ordinal regression. In Section 4, we compare several classification models for ordinal regression in an unified view. In Section 5, we described a generic algorithm for learning the proposed models. Experiments are presented in Section 6 and Section 7 concludes the paper.

2 Ordinal Regression as Linear Multi-class Classification

Let us consider one-dimensional observations $x \in \mathcal{X} = \mathbb{R}$ in which case the ORD classifier $h(x) = 1 + \sum_{k=1}^{Y-1} [\![x > \theta_k]\!]$ splits the real axis into Y intervals defined by thresholds $\theta_1 \leq \theta_2 \leq \cdots \leq \theta_{Y-1}$. One may think of representing the ORD classifier in the form

$$h'(x) = \operatorname*{argmax}_{y \in \mathcal{Y}} f(x, y) \,, \tag{2}$$

where $f \colon \mathbb{R} \times \mathcal{Y} \to \mathbb{R}$ is a discriminant function. If we manage to construct the discriminant functions such that $f(x, y) \geq f(x, y')$, $y' \in \mathcal{Y} \setminus \{y\}$ iff $h(x) = y$ then both representations will be equivalent i.e. $h'(x) = h(x)$, $x \in \mathbb{R}$. Let us consider

a linear discriminant function with the slope equal to y, i.e. $f(x, y) = x \cdot y + b_y$, in which case (2) becomes a linear multi-class classifier. It is not difficult to see that such linear classifier also splits the real axis into intervals. Fig 1 shows an example of the ORD classifier and its equivalent linear classifier $h'(x)$.

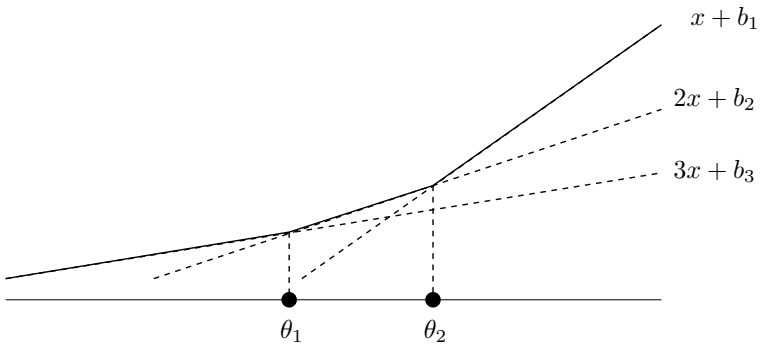

Fig. 1. The figure illustrates relation between the ORD classifier $h(x) = 1 + \sum_{k=1}^{Y-1} [\![x > \theta_k]\!]$ and its alternative representation $h'(x) = \operatorname{argmax}_{y \in \mathcal{Y}}(x \cdot y + b_y)$ for the $(Y = 3)$-class problem. Note, that x and y-axes have different scale in order to save space.

The same idea can be applied for n-dimensional observations $\boldsymbol{x} \in \mathcal{X} = \mathbb{R}^n$. The multi-class linear classifier which can represent the ORD classifier (1) reads

$$h'(\boldsymbol{x}; \boldsymbol{w}, \boldsymbol{b}) = \operatorname*{argmax}_{y \in \mathcal{Y}} \Big(\langle \boldsymbol{x}, \boldsymbol{w} \rangle \cdot y + b_y \Big) , \tag{3}$$

where $\boldsymbol{w} \in \mathbb{R}^n$ is parameter vector and $\boldsymbol{b} = (b_1, \ldots, b_Y) \in \mathbb{R}^Y$ is a vector of intercepts. We denote (3) as the Multi-class ORDinal (MORD) classifier. Inside the paper we assume that the "argmax" operator returns the minimal label in the case of more than one maximizer.

A natural question is whether both representations are equivalent in the sense that any ORD classifier can be represented by some MORD classifier and vice-versa. The following theorem gives a positive answer to the question.

Theorem 1. *The ORD classifier (1) and the MORD classifier (3) are equivalent in the following sense. For any $\boldsymbol{w} \in \mathbb{R}^n$ and admissible $\boldsymbol{\theta} \in \Theta$ there exists $\boldsymbol{b} \in \mathbb{R}^Y$ such that $h(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{\theta}) = h'(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{b})$, $\forall \boldsymbol{x} \in \mathbb{R}^n$. For any $\boldsymbol{w} \in \mathbb{R}^n$ and $\boldsymbol{b} \in \mathbb{R}^n$ there exists admissible $\boldsymbol{\theta} \in \Theta$ such that $h(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{\theta}) = h'(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{b})$, $\forall \boldsymbol{x} \in \mathbb{R}^n$.*

A proof is given in Appendix A.

Our proof is constructive in the sense that we can provide a conversion from the ORD classifier to the MORD classifier and vice-versa. In exotic cases, which however may appear in practice, some classes can collapse to a single point and effectively disappear. To cover all such situations, we first define the concept of non-degenerated classifier and then we give formulas for the conversions.

Definition 1 (Degenerated and non-degenerated classifier). *We call class $y \in \mathcal{Y}$ non-degenerated for classifier $h'(\boldsymbol{x})$ iff $\mathcal{X}_y = \text{interior}(\{\boldsymbol{x} \in \mathcal{X} : h'(\boldsymbol{x}) = y\}) \neq \emptyset$. Classifier $h'(\boldsymbol{x})$ is non-degenerated iff all classes are non-degenerated. In opposite case the classifier is called degenerated.*

Given a MORD classifier, the class $\hat{y} \in \mathcal{Y}$ is non-degenerated iff the linear inequalities

$$\begin{aligned} z\hat{y} + b_{\hat{y}} &> z(\hat{y} - k) + b_{\hat{y}-k},\ 1 \leq k < \hat{y}\,, \\ z\hat{y} + b_{\hat{y}} &\geq z(\hat{y} + t) + b_{\hat{y}+k},\ 1 < t \leq Y - \hat{y}\,, \end{aligned} \tag{4}$$

are solvable w.r.t. $z \in \mathbb{R}$. It is seen that we can check it in $\mathcal{O}(Y)$ time. We refer to the proof for more details.

Conversion formulas. Given parameters of the ORD classifier $\boldsymbol{w} \in \mathbb{R}^n$, $\boldsymbol{\theta} \in \Theta$, the equivalent MORD classifier has parameters $\boldsymbol{w}$ and $\boldsymbol{b}$ given by

$$b_1 = 0 \qquad \text{and} \qquad b_y = -\sum_{i=1}^{y-1} \theta_i,\ y = 2, \ldots, Y. \tag{5}$$

The conversion from the MORD classifier to the ORD classifier is done differently for the non-generated and the degenerated classifier. Given parameters of a non-degenerated MORD classifier $\boldsymbol{w} \in \mathbb{R}^n$ and $\boldsymbol{b} \in \mathbb{R}^Y$, we can compute thresholds $\boldsymbol{\theta} \in \Theta$ of the equivalent ORD classifier by

$$\theta_y = b_y - b_{y+1}, \qquad y = 1, \ldots, Y - 1\,. \tag{6}$$

Given parameters of a degenerated MORD classifier $\boldsymbol{w} \in \mathbb{R}^n$ and $\boldsymbol{b} \in \mathbb{R}^Y$, we compute thresholds $\boldsymbol{\theta} \in \Theta$ of the equivalent ORD classifier by

$$\theta_{y_i} = \cdots = \theta_{y_{i+1}-1} = \tfrac{b_{y_i} - b_{y_{i+1}}}{(y_{i+1} - y_i)},\ i = 1, \ldots, p, \tag{7}$$

where $y_i \in \mathcal{Y},\ i = 1, \ldots, p$ is an increasing subsequence of non-degenerated classes.

Finally, let us note that the MORD classifier is represented by $n + Y$ parameters insted of $n + Y - 1$ parameters of the ORD classifier. However, the parameters of the MORD classifier are unconstrained which makes the MORD representation attractive for learning because no additional constraints on the intercepts $\boldsymbol{\theta} \in \Theta$ are not needed.

3 Piece-wise Ordinal Regression Classifier

The discriminative power of the ORD classifier can be limiting in some cases. Mapping the observations into higher dimensional space via usage of kernel functions is one way to make the linear ORD classifier more discriminative. Though the "kernalization" of the ORD classifier is straightforward it is not suitable in all cases. For example, the kernels are prohibitive in applications which require processing of large amounts of training examples and/or if a real-time response

of the classifier is the must. Instead, we proposed to stay in the original feature space where we construct a combined classifier from a set of simpler component classifiers. In our case, the component classifiers will be the MORD classifiers, each responsible for a subset of labels.

Let $Z > 1$ be a number of cut labels $(\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_Z) \in \mathcal{Y}^Z$ such that $\hat{y}_1 = 1$, $\hat{y}_Z = Y$ and $\hat{y}_z \leq \hat{y}_{z+1}$, $z \in \mathcal{Z} = \{1, \ldots, Z-1\}$. The cut labels define a partitioning of $\mathcal{Y}$ into Z subsets $\mathcal{Y}_z = \{y \in \mathcal{Y} \mid \hat{y}_z \leq y \leq \hat{y}_{z+1}\}$, $z \in \mathcal{Z}$. We will model dependence between the observation $\boldsymbol{x}$ and a subset of labels $\mathcal{Y}_z$ by a component classifier

$$h_z(\boldsymbol{x}) = \operatorname*{argmax}_{y \in \mathcal{Y}_z} f_z(\boldsymbol{x}, y) \tag{8}$$

where $f_z \colon \mathbb{R}^n \times \mathcal{Y}_z \to \mathbb{R}$ is a discriminant function. We define a combined classifier whose discriminant function is composed of discriminant functions of the component classifiers as follows

$$h''(\boldsymbol{x}) = \operatorname*{argmax}_{z \in \mathcal{Z}} \max_{y \in \mathcal{Y}_z} f_z(\boldsymbol{x}, y)\,. \tag{9}$$

We set the discriminant functions to be

$$f_z(\boldsymbol{x}, y) = \big\langle \boldsymbol{x}, \boldsymbol{w}_z(1 - \alpha(y, z)) + \boldsymbol{w}_{z+1}\alpha(y, z) \big\rangle + b_y \tag{10}$$

where

$$\alpha(y, z) = \frac{y - \hat{y}_z}{\hat{y}_{z+1} - \hat{y}_z}$$

and $\boldsymbol{W} = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_Z] \in \mathbb{R}^{n \times Z}$, $\boldsymbol{b} \in \mathbb{R}^Y$ are parameters. With these definitions it can be shown that: i) the component classifiers (8) are the ORD classifiers and ii) the combined classifier (9) is well defined because all its neighboring discriminant functions are consistent at the cut labels, i.e. $f_z(\boldsymbol{x}, \hat{y}_{z+1}) = f_{z+1}(\boldsymbol{x}, \hat{y}_{z+1})$, $z \in \mathcal{Z}$, holds. The claim i) is seen after substituting (10) into (8) which after some algebra yields

$$h_z(\boldsymbol{x}) = \operatorname*{argmax}_{y \in \mathcal{Y}_z} \Big(\langle \boldsymbol{x}, \boldsymbol{w}_{z+1} - \boldsymbol{w}_z \rangle \alpha(y, z) + b_y \Big)$$

and since $\alpha(y, z)$ is linearly increasing with y, Theorem 1 guarantees that $h_z(\boldsymbol{x})$ is the MORD classifier equivalent to the ORD classifier. The claim ii) follows from the fact that $\alpha(\hat{y}_{z+1}, z) = 1$ and $\alpha(\hat{y}_{z+1}, z+1) = 0$, and thus $f_z(\boldsymbol{x}, \hat{y}_{z+1}) = \langle \boldsymbol{x}, \boldsymbol{w}_{z+1} \rangle + b_{\hat{y}_{z+1}} = f_{z+1}(\boldsymbol{x}, \hat{y}_{z+1})$.

We can explicitly write the component classifier, which we call the Piece-Wise Multi-class ORDinal (PW-MORD) classifier, as follows

$$h''(\boldsymbol{x}, \boldsymbol{W}, \boldsymbol{b}) = \operatorname*{argmax}_{z \in \mathcal{Z}} \operatorname*{argmax}_{y \in \mathcal{Y}_Z} \Big(\langle \boldsymbol{x}, \boldsymbol{w}_z(1 - \alpha(y, z)) + \boldsymbol{w}_{z+1}\alpha(y, z) \rangle + b_y \Big)\,. \tag{11}$$

Figure 2 visualizes the ORD (=MORD) and the PW-MORD classifier on a toy data. It is seen that the distribution of the data cannot be well described by the ORD classifier while the PW-MORD composed of 3 ORD classifiers provides much better model in this case.

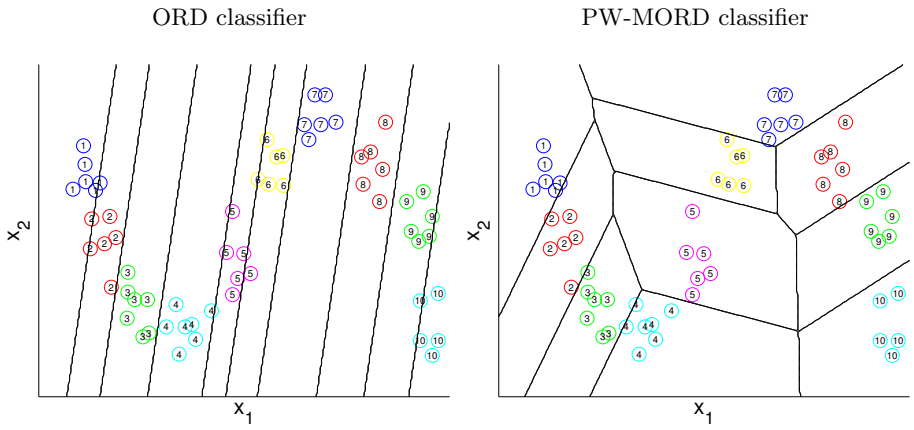

Fig. 2. The figure shows the partitioning of 2-dimensional feature space realized by the ORD classifier and the PW-MORD classifier with $Z = 3$ components. The cut labels for the PW-MORD classifier where set to $\{1, 4, 7, 10\}$.

4 Unified View of Classifiers for Ordinal Regression

Let us consider the linear multi-class classifier

$$h(\boldsymbol{x}, \boldsymbol{W}, \boldsymbol{b}) = \operatorname*{argmax}_{y \in \mathcal{Y}} \left(\langle \boldsymbol{x}, \sum_{z=1}^{Z} \beta(y, z) \boldsymbol{w}_z \rangle + b_y \right) \tag{12}$$

where $\boldsymbol{W} = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_Z] \in \mathbb{R}^{n \times Z}$, $\boldsymbol{b} = [b_1; \ldots; b_Y] \in \mathbb{R}^Y$ are parameters and $\beta \colon \mathcal{Y} \times \{1, \ldots, Z\} \to \mathbb{R}$ are fixed numbers. We are going to describe several instances of the classifier (12) which can be useful models for ordinal regression.

1. Rounded linear-regression rule

$$h(\boldsymbol{x}, \boldsymbol{w}, b) = \max(1, \min(Y, \operatorname{round}(\langle \boldsymbol{w}, \boldsymbol{x} \rangle + b))) \tag{13}$$

is the most simplest model for the ordinal regression obtained by clipping a rounded response of the standard linear regression rule to the interval $[1, Y]$. It is easy to show that (13) is an instance of (12) recovered after setting $Z = 1$, $\beta(1, y) = 2y$, $y \in \mathcal{Y}$, and fixing the components of the intercept vector $\boldsymbol{b}$ to $b_y = 2by - y^2$. Using the conversion formula (6) we can show that the rounded linear-regression rule is equivalent to the ORD classifier with equal width of the decision intervals, namely, with $\theta_{k+1} - \theta_k = 2$, $k = 1, \ldots, Y - 2$.

2. Multi-class linear classifier

$$h(\boldsymbol{x}, \boldsymbol{W}, \boldsymbol{b}) = \operatorname*{argmax}_{y \in \mathcal{Y}} \left(\langle \boldsymbol{w}_y, \boldsymbol{x} \rangle + b_y \right) \tag{14}$$

is recovered after setting $Z = Y$ and $\beta(y, z) = [\![y = z]\!]$, $y \in \mathcal{Y}$, $z \in \{1, \ldots, Z\}$. It is the most generic (and also most discriminative) form of (12) which completely ignores ordering of the labels.

3. The proposed MORD classifier (3) is recovered after setting $Z = 1$, $\boldsymbol{W} = \boldsymbol{w}_1$, and $\beta(y, 1) = y$, $y \in \mathcal{Y}$. We showed that the MORD classifier is equivalent to the standard ORD classifier (1) most frequently used in the ordinal regression.

4. The proposed PW-MORD classifier (11) is recovered after setting $\beta(y, z)$ according to

$$\begin{array}{ll} \beta(y,z) = 1 - \alpha(y,z) & \text{for } z = 1, \ldots, Z-1\,, y \in \mathcal{Y}_z\,, \\ \beta(y,z) = \alpha(y,z-1) & \text{for } z = 2, \ldots, Z\,, y \in \mathcal{Y}_z\,, \\ \beta(y,z) = 0 & \text{otherwise.} \end{array} \tag{15}$$

The PW-MORD is composed from $Z - 1$ MORD classifiers each modeling a subset of labels (see Section 3). The PW-MORD is most flexible as it allows to smoothly control its the complexity by a single parameter Z. It is easy to see that for $Z = 2$ the PW-MORD is equivalent to the MORD (=ORD) classifier while for $Z = Y$ it becomes the Multi-class linear classifier.

5 Generic Learning Algorithm for Ordinal Regression

In this section we consider problem of learning parameters of the generic linear multi-class classifier (12) from given example set $\{(\boldsymbol{x}^1, y^1), \ldots, (\boldsymbol{x}^m, y^m)\} \in (\mathbb{R}^n \times \mathcal{Y})^m$. We propose to use a generic and well understood framework originally developed for the structured output learning [8]. Following [8], we define an approximate empirical risk

$$R(\boldsymbol{W}, \boldsymbol{b}) = \frac{1}{m} \sum_{i=1}^{m} \max_{y \in \mathcal{Y}} \left[\Delta(y, y^i) + \langle \boldsymbol{x}^i, \sum_{z \in \mathcal{Z}} \beta(y, z) \boldsymbol{w}_z \rangle (y - y^i) + b_y - b_{y^i} \right], \tag{16}$$

where $\Delta \colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ is any loss function satisfying

$$\Delta(y, y) = 0\,, \forall y \in \mathcal{Y} \quad \text{and} \quad \Delta(y, y') > 0\,, \forall (y, y') \in \mathcal{Y}^2 \quad \text{such that} \quad y \neq y'\,. \tag{17}$$

This risk approximation uses the idea of the margin-rescaling loss functions [8] applied on the classifier (12). It is easy to prove that $R(\boldsymbol{w}, \boldsymbol{b})$ is a convex upper bound on the true empirical risk

$$R_{\text{emp}}(\boldsymbol{W}, \boldsymbol{b}) = \frac{1}{m} \sum_{i=1}^{m} \Delta(y^i, h(\boldsymbol{x}^i, \boldsymbol{W}, \boldsymbol{b}))\,.$$

We can formulate learning of the classifier (12) as the following convex unconstrained minimization problem

$$(\boldsymbol{W}^*, \boldsymbol{b}^*) = \underset{\boldsymbol{W} \in \mathbb{R}^n, \boldsymbol{b} \in \mathbb{R}^Y}{\operatorname{argmin}} \left[\frac{\lambda}{2} \Big(\|\boldsymbol{W}\|^2 + \|\boldsymbol{b}\|^2 \Big) + R(\boldsymbol{W}, \boldsymbol{b}) \right]. \tag{18}$$

where $\|\cdot\|$ denotes the Frobenius norm and $\lambda > 0$ is a prescribed (regularization) constant used to control over-fitting. The setting with $\lambda = 0$, referred to as the

empirical risk minimization learning, means that we simply try to find the classifier with minimal upper bound $R(\boldsymbol{W}, \boldsymbol{b})$ on the empirical risk $R_{\text{emp}}(\boldsymbol{W}, \boldsymbol{b})$, in other words the one which performs best on training examples measured in terms of the prescribed loss $\Delta(y, y')$. The setting $\lambda > 0$, referred to as the regularized risk minimization learning, is equivalent to minimizing the risk $R(\boldsymbol{W}, \boldsymbol{b})$ w.r.t. parameters constrained to be inside a ball with radius inversely proportional to λ. The latter setting can be also interpreted as trying to maximize a generalized margin between the training examples and the classifier.

A big effort has been put by the ML community into development of efficient solvers for the problem (18). For example, a generic bundle method for risk minimization [9] or its accelerated variant [10] can be readily applied to solve (18).

Let us compare our framework with the existing algorithms for learning the ORD classifier. The existing algorithms consider a limited set of loss functions $\Delta(y, y')$. The most generic approach of [6] derives an upper bound for so called V-shaped losses: a loss is called V-shaped if it satisfies (17) and in addition

$$\Delta(y, y') \geq \Delta(y, y'+1) \quad \text{if} \quad y' \leq y \quad \text{and} \quad \Delta(y, y') \leq \Delta(y, y'+1) \quad \text{if} \quad y' \geq y\,. \tag{19}$$

The V-shaped loss (19) subsumes the most frequently used losses, i.e. the MAE loss $\Delta(y, y') = |y - y'|$ and the 0/1-loss $\Delta(y, y') = [\![y \neq y]\!]$, yet it is not as generic as the loss (17) applicable in our framework. Next limitation of the existing algorithms is that they have to care about feasibility of the thresholds $\boldsymbol{\theta} \in \Theta$ because they work directly on the parameters of the ORD classifier. This requires to either introduce additional constraints on the thresholds $\boldsymbol{\theta} \in \Theta$ or to impose additional constraints on the loss function, namely, that the loss must be convex [6]. For instance, the 0/1-loss is not convex hence the learning algorithms require extra inequality constraints (like the SVOR-EXP algorithm of [5]) which may complicate the optimization. Note that in our approach the problem (18) remains unconstrained irrespectively to the selected loss.

The generality of our framework, however, does not automatically imply that the risk approximation (16) is better (tighter) than those used in existing methods. We experimentally show that in the case of the MAE loss, i.e. the most frequently used one, the proposed approximation (16) provides slightly but consistently better approximation than the existing ones.

6 Experiments

In this section we empirically compare the proposed methods with existing algorithms. In Section 6.1 we present experiments on standard benchmarks. Experiments on real-life problem of visual age estimation are described in Section 6.2.

In our experiments we compare the following methods:

1. MORD. Proposed classifier (3) trained by (18) using the MAE loss.
2. PW-MORD. Proposed classifier (11) trained by (18) using the MAE loss.

3. LinReg. Rounded linear regressor (13) trained by (18) using the MAE loss.
4. LinClass. Standard multi-class linear classifier (14) trained by (18) using the MAE loss. It is an instance of the Structured Output Classifier [8]. Note that LinClass is up to the loss very similar to the standard multi-class SVM.
5. SVOR-EXP. Support Vector Ordinal Regression with explicit constraints [5].
6. SVOR-IMC. Support Vector Ordinal Regression with implicit constraints [5].

The SVOR-IMC and SVOR-EXP are instances of a generic framework of [6] developed for learning the ORD classifier (1). It was shown that the algorithms minimize a convex upper bound on the MAE-loss (SVOR-IMC) and the 0/1-loss (SVOR-EXP), respectively. Other methods for learning the ORD classifier have been proposed like SVM-based algorithms of [4] or the Support Vector Regression [2]. However, they are consistently outperformed by the SVOR-EXP and SVOR-IMC hence we compare only against the latter two.

We consider the MAE $\Delta(y, y') = |y - y'|$ as the desired metric because it is by far the most frequently used loss in the ordinal regression context as well as it is suitable for the real-life problem we consider.

All tested algorithms are instances of the regularized risk minimization framework (18). Note that SVOR-IMC and SVOR-EXP were originally formulated as quadratic programs but can be easily converted to (18). In the case of SVOR-IMC the problem (18) uses additional constraints $\boldsymbol{\theta} \in \Theta$. The learning problem (18) is specified up to the regularization constant λ tuned on validation data from a fixed set of values Λ. In particular, we set $\Lambda = \{1, 0.1, 0.01, 0.001, 0\}$. We used two optimization algorithms to solve (18). For $\lambda > 0$ we used the Bundle Methods for Risk Minimization (BMRM) [9]. For $\lambda = 0$ we used the Analytic Center Cutting Plane Method (ACCPM) proposed in [11]. To avoid implementation bias, we wrote all algorithms by ourselves using mainly Matlab and C only to program a QP solver called inside the BMRM algorithm. In both cases we set the solvers to find the ε-optimal solution of the learning objective, in particular, we stopped the solver if the objective was below factor of 1.01 of the optimal value (we use the Lagrange dual to get the optimality certificate).

6.1 Standard Benchmarks

We performed experiments with seven data sets [1] used in [5][6]. We followed exactly the same evaluation protocol. The data were produced by discretising metric regression problems into $Y = 10$ bins. Data are randomly partitioned to train/test part. The partitioning was repeated 20 times. The features are normalized to have zero mean and unit variance coordinate wise. The reported results are averages and standard deviations computed over the 20 partitions. The feature dimension and train/test ratios are listed in Table 1.

We performed two experiments. The goal of the first experiment is to assess the ability of the proposed training algorithm (18) to minimize the empirical

[1] The link `http://www.dcc.fc.up.pt/~ltorgo/Regression/census.tar.gz` to the eight dataset "Census" was broken hence we could not include it.

	n	train/test	TrnRisk	MORD	SVOR-IMC	SVOR-EXC
Pyrimidines	27	50/24	MAE	**0.433 (0.093)**	0.482 (0.104)	0.491 (0.125)
			0/1	0.343 (0.064)	0.391 (0.069)	**0.329 (0.078)**
MachineCPU	6	150/59	MAE	**0.914 (0.052)**	0.920 (0.046)	0.972 (0.068)
			0/1	0.602 (0.035)	0.611 (0.027)	**0.594 (0.029)**
Boston	13	300/206	MAE	**0.812 (0.043)**	0.823 (0.047)	0.869 (0.050)
			0/1	0.558 (0.026)	0.573 (0.027)	**0.551 (0.028)**
Abalone	8	1000/3177	MAE	**1.412 (0.038)**	1.422 (0.041)	1.632 (0.063)
			0/1	0.734 (0.015)	0.748 (0.017)	**0.715 (0.016)**
Bank	32	3000/5192	MAE	**1.421 (0.021)**	1.429 (0.021)	1.913 (0.051)
			0/1	0.700 (0.006)	0.716 (0.007)	**0.690 (0.005)**
Computer	21	4000/4192	MAE	0.632 (0.010)	**0.632 (0.010)**	0.653 (0.012)
			0/1	**0.477 (0.006)**	0.480 (0.006)	0.477 (0.008)
California	8	5000/15640	MAE	**1.178 (0.013)**	1.182 (0.014)	1.233 (0.014)
			0/1	0.692 (0.008)	0.697 (0.007)	**0.681 (0.008)**

Table 1. Comparison of the MORD, SVOR-IMC and SVOR-EXC in terms of the ability to minimize the empirical risk measured in terms of the MAE and 0/1-loss. The columns 2 and 3 show data dimension and training/testing split ratio, respectively.

risk if compared to the existing algorithms SVOR-IMC and SVOR-EXC. Note that all the tested methods learn exactly the same ORD classifier by minimizing a convex approximation of the empirical risk whose direct optimization is not tractable. SVOR-IMC and SVOR-EXC use a specific risk approximation tailored for the ORD classifier. Our method makes it possible to train the ORD classifier using the standard margin-rescaling. In this experiment we set $\lambda = 0$, i.e. we minimized just the risk approximation which we want to assess. Table 1 summarizes the results. It is seen that our method slightly but consistently (up to one near draw for "Computer" data) outperforms the SVOR-IMC approximation in terms of the MAE loss for which both methods were intended. The results of SVOR-EXC optimizing the 0/1-loss are included just for completeness.

The goal of the second experiment is to assess the ability to minimize the test risk (generalization error). We compare the proposed methods PW-MORD against the standard models. We considered the PW-MORD with $Z = 2, 3, 4$ and the cut labels set symmetrically, i.e. $\{1, 10\}$, $\{1, 5, 10\}$ and $\{1, 4, 7, 10\}$. Note that PW-MORD with Z=2 corresponds to the MORD classifier hence not included in testing. The optimal complexity of the PW-MORD classifier, i.e. the number Z, as well as the regularization constant $\lambda \in \{1, 0.1, 0.01, 0.001, 0\}$ for all methods were selected based on 5-fold cross-validation estimate of the MAE (on training split). Table 2 summarizes the results. In most cases the PW-MORD classifier outperformed the competitors in terms of the target MAE metric. We attribute this fact to its flexible complexity and the ability of the proposed training algorithm to well approximate the loss function (see previous experiment). The PW-MORD was outperformed only by the LinReg on the "Pyrimids" data and by the LinCls on the "California" data. This is result is not surprising because the "Pyrimids" data have very few training examples hence the simplest regres-

sion model best avoids over-fitting. On the other hand, the "California" data are low dimensional with high number of training examples and thus LinCls, the most flexible model, can best describe the data without overfitting, i.e. the ordering prior imposed by the other models is not needed here. A surprising result is that the winner in terms of the MAE loss is in most cases the best method for the 0/1-loss, i.e. it is better than the SVOR-EXC algorithm directly optimizing the 0/1-loss. Currently we do not have a good explanation of this observation.

	TstRisk	LinCls	LinReg	PW-MORD	(Z)	SVOR-IMC	SVOR-EXC
Pyrimidines	MAE	1.59 (0.25)	**1.37 (0.27)**	1.50 (0.38)	4	1.52 (0.29)	1.63 (0.28)
	0/1	0.76 (0.10)	0.76 (0.10)	**0.74 (0.09)**		0.79 (0.07)	0.80 (0.08)
MachineCPU	MAE	1.00 (0.15)	1.03 (0.10)	**0.95 (0.12)**	2	0.95 (0.11)	1.01 (0.13)
	0/1	0.65 (0.06)	0.70 (0.06)	**0.62 (0.06)**		0.63 (0.06)	0.65 (0.05)
Boston	MAE	0.94 (0.07)	0.95 (0.06)	**0.86 (0.05)**	3	0.91 (0.06)	0.97 (0.08)
	0/1	0.62 (0.03)	0.64 (0.03)	**0.58 (0.03)**		0.61 (0.03)	0.62 (0.04)
Abalone	MAE	1.42 (0.02)	1.51 (0.01)	**1.41 (0.02)**	4	1.47 (0.01)	1.68 (0.04)
	0/1	0.73 (0.01)	0.79 (0.01)	**0.73 (0.01)**		0.76 (0.01)	0.73 (0.01)
Bank	MAE	1.45 (0.01)	1.51 (0.01)	**1.45 (0.01)**	4	1.45 (0.01)	1.94 (0.05)
	0/1	0.70 (0.01)	0.77 (0.01)	0.70 (0.01)		0.72 (0.01)	**0.69 (0.00)**
Computer	MAE	0.62 (0.01)	0.72 (0.01)	**0.61 (0.01)**	4	0.63 (0.01)	0.65 (0.01)
	0/1	0.47 (0.00)	0.56 (0.01)	**0.47 (0.01)**		0.48 (0.01)	0.48 (0.00)
California	MAE	**1.12 (0.00)**	1.21 (0.01)	1.14 (0.00)	4	1.18 (0.01)	1.23 (0.01)
	0/1	**0.67 (0.00)**	0.71 (0.00)	0.68 (0.00)		0.70 (0.00)	0.68 (0.00)

Table 2. Comparison of various classification models in terms of the test risk measured in terms of the MAE and the 0/1-loss. The column (Z) shows the best complexity of the PW-MORD classifier selected in the cross-validation stage.

6.2 Visual Age Prediction

We consider problem of predicting an apparent age of a person from an image of his/her face. We experimented on a dataset containing 37,668 face images obtained by putting together standard face-recognition benchmarks (Feret, PAL, LFW, BioID, FaceTracer, xm2vts) and completing the rest by images downloaded from the Internet. Images were manually annotated by age which was in range from 0 to 100 years. The database has equal ratio of male and females. Each face was registered by a landmark detector [12], normalized to a canonical image 30×20 by affine transform and described by pyramid-of-LBP descriptor [13]. Each face is represented by $n = 159,488$-dimensional sparse binary vector. We randomly split the data into training/validation/test part in ration 60/20/20. The validation part is used to tune the regularization constant. The reported results are averages and standard deviations of test errors computed over 3 splits.

We compared the linear multi-class SVM classifier (LinCls), the standard ordinal regression model implemented via the MORD representation and the

PW-MORD classifier. In the case of LinCls we had to discretize the age into 10 equal bins because modeling all 101 classes would not be feasible (only representation of the classifier would require 120MB). We used PW-MORD with Z=11 and set the cut labels to equally cover the range of 101 years. Thus the PW-MORD classifier models each decade by a single linear ordinal regression classifier. We also compared against a commercial face recognition system developed by FACE.COM [2]. Results are summarized in Table 3 reporting the target MAE loss as well as the error levels. The proposed PW-MORD significantly outperformed all competing ordinal regression models by significant margin. The MORD classifier (=standard ORD model) is apparently not sufficiently discriminative. On the other hand, training full multi-class classier for all 101 classes is not feasible. The PW-MORD model also compares favorably with the FACE.COM system. Namely, in terms of MAE metric the FACE.COM is slightly better, however, the PW-MORD provides substantially better results for lower error levels what is typically preferred in practice.

TstRisk	LinCls	PW-MORD	MORD	FaceCom
MAE	11.19 (0.16)	7.92 (0.06)	14.53 (0.13)	7.89 (NA)

	Occurrence in [%]			
Error level	LinCls	PW-MORD	MORD	FaceCom
5	44.8	52.9	28.3	47.4
10	65.2	74.5	47.8	73.7
20	84.6	91.1	74.4	93.3
30	91.7	97.2	88.9	98.2
40	94.8	99.2	95.6	99.5
50	96.4	99.8	98.4	99.9
60	97.8	99.9	99.6	100.0
70	98.9	100.0	99.9	100.0

Table 3. Comparison of various classifiers on the visual age estimation problem. The upper table shows the test MAE, i.e. average prediction error in years. The bottom table shows error levels for the tested classifiers, e.g. the first row tells the percentage of examples with MAE not greater than 5 years.

7 Conclusions

We have shown equivalence between the classification rule used in ordinal regression and a class of linear multi-class classifiers. The established equivalence has the following benefits. First, it allows to better understand various classification

[2] FACE.COM (`www.face.com`) provided a free access server with face recognition technology. We passed our data though the server between July 15 and August 15, 2012. Recently, the company has been acquired by Facebook and the server closed.

models. Second, it provides a path to develop new learning algorithms for ordinal regression borrowing from well understand multi-class classification. Third, it allows to design new models for ordinal regression with higher discriminative power. Experiments on standard benchmarks as well as a real-life problem of visual age estimation demonstrate usefulness of the proposed method.

Acknowledgments. KA was supported by the Grant agency of CTU project SGS12187/OHK3/3T/13 and the Visegrad Scholarship contract No. 51200430, VF by the Grant Agency of the Czech Republic under Project P202/12/2071, VH by EC project FP7-288533 CLOPEMA and the Technology Agency of the Czech Republic under Project TE01020197.

References

1. McCullagh, P.: Regression models for ordinal data. Journal of the Royal Statical Society **42**(2) (1980) 109–142
2. Vapnik, V.N.: Statistical learning theory. Adaptive and Learning Systems. Wiley, New York, New York, USA (1998)
3. Crammer, K., Singer, Y.: Pranking with ranking. In: Advances in Neural Information Processing Systems (NIPS). (2001) 641–647
4. Shashua, A., Levin, A.: Ranking with large margin principle: Two approaches. In: Proceedings of Advances in Neural Information Processing Systems (NIPS). (2002)
5. Chu, W., Keerthi, S.S.: New approaches to support vector ordinal regression. In: Proc. of the International conference on Machine Learning (ICML). (2005) 145–152
6. Li, L., Lin, H.T.: Ordinal regression by extended binary classification. In: Advances in Neural Information Processing Systems (NIPS). (2006)
7. Fen, X., Liang, Z., Y.Yang, W.Zhang: Ordinal regression as multiclass classification. Intern. Journal of Intelligent Control and Systems **12**(3) (2007) 230–236
8. Tsochantaridis, I., Joachims, T., Hofmann, T., Altun, Y., Singer, Y.: Large margin methods for structured and interdependent output variables. Journal of Machine Learning Research **6** (2005) 1453–1484
9. Teo, C.H., Vishwanthan, S., Smola, A.J., Le, Q.V.: Bundle methods for regularized risk minimization. Journal of Machine Learning Research **11** (2010) 311–365
10. Franc, V., Sonneburg, S.: Optimized cutting plane algorithm for large-scale risk minimization. Journal of Machine Learning Research **10** (2009) 2157–2232
11. Antoniuk, K., Franc, V., Hlaváč, V.: Learning markov networks by analytic center cutting plane method. In: Proceedings of International Conference on Pattern Recognition (ICPR). (2012) 2250–2253
12. Uřičář, M., Franc, V., Hlaváč, V.: Detector of facial landmarks learned by the structured output SVM. In: Proceedings of the International Conference on Computer Vision Theory and Applications (VISAPP). (2012) 547–556
13. Sonnenburg, S., Franc, V.: Coffin: A computational framework for linear svms. In: Proceedings of the 27th Annual International Conference on Machine Learning (ICML 2010), Madison, USA, Omnipress (June 2010)

Appendix A: Proof of Theorem 1

Let us prove the first part of theorem stating that for any $\boldsymbol{w} \in \mathbb{R}^n$ and admissible $\boldsymbol{\theta} \in \Theta$ there exists $\boldsymbol{b} \in \mathbb{R}^Y$ such that $h(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{\theta}) = h'(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{b})$, $\forall \boldsymbol{x} \in \mathbb{R}^n$. In particular we show that $\boldsymbol{b} \in \mathbb{R}^Y$ given by the formula (5) satisfies theorem.

First, suppose the ORD classifier $h(\boldsymbol{x}; \boldsymbol{w}, \boldsymbol{\theta})$ outputs $y \in \mathcal{Y}$ for some $\boldsymbol{x} \in \mathcal{X}$, i.e. $\theta_y \geq \langle \boldsymbol{w}, \boldsymbol{x} \rangle > \theta_{y-1}$ holds[1]. The MORD classifier $h'(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{b})$ outputs the same y iff the system of inequalities

$$\begin{aligned} &\langle \boldsymbol{w}, \boldsymbol{x} \rangle y + b_y > \langle \boldsymbol{w}, \boldsymbol{x} \rangle (y-k) + b_{y-k}, \ 1 \leq k < y, \\ &\langle \boldsymbol{w}, \boldsymbol{x} \rangle y + b_y \geq \langle \boldsymbol{w}, \boldsymbol{x} \rangle (y+t) + b_{y+t}, \ 1 \leq t \leq Y - y \end{aligned} \tag{20}$$

holds. The system (20) can be rewritten as[2]

$$\begin{aligned} &\langle \boldsymbol{w}, \boldsymbol{x} \rangle k > \sum_{i=y-k}^{y-1} \theta_i, \ 1 \leq k < y, \\ &\langle \boldsymbol{w}, \boldsymbol{x} \rangle t \leq \sum_{i=y}^{y+t-1} \theta_i, \ 1 \leq t \leq Y - y. \end{aligned} \tag{21}$$

The validity of (21) follows from

$$\begin{aligned} &\langle \boldsymbol{w}, \boldsymbol{x} \rangle k > \theta_{y-1} k \geq \sum_{i=y-k}^{y-1} \theta_i, \ 1 \leq k < y, \\ &\langle \boldsymbol{w}, \boldsymbol{x} \rangle t \leq \theta_y t \leq \sum_{i=y}^{y+t-1} \theta_i, \ 1 \leq t \leq Y - y \ , \end{aligned} \tag{22}$$

where the first inequality (on both lines) is induced by $\theta_y \geq \langle \boldsymbol{w}, \boldsymbol{x} \rangle > \theta_{y-1}$ and the second inequality (also on both lines) is due to $\theta_1 \leq \theta_2 \leq \cdots \leq \theta_{Y-1}$.

Second, suppose the MORD classifier $h'(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{b})$ outputs $y \in \mathcal{Y}$ for some $\boldsymbol{x} \in \mathcal{X}$, which means that

$$\begin{aligned} &\langle \boldsymbol{w}, \boldsymbol{x} \rangle y + b_y > \langle \boldsymbol{w}, \boldsymbol{x} \rangle (y-1) + b_{y-1}, \\ &\langle \boldsymbol{w}, \boldsymbol{x} \rangle y + b_y \geq \langle \boldsymbol{w}, \boldsymbol{x} \rangle (y+1) + b_{y+1}, \end{aligned} \tag{23}$$

which is equivalent to

$$b_y - b_{y+1} \geq \langle \boldsymbol{w}, \boldsymbol{x} \rangle > b_{y-1} - b_y \ . \tag{24}$$

Finally, after combining (24) with (5) we obtain $\theta_y \geq \langle \boldsymbol{w}, \boldsymbol{x} \rangle > \theta_{y-1}$, which implies that the ORD classifier $h(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{\theta})$ outputs the same y.

Let us make an observation before proving the second part of the theorem. Let $y_1, \ldots, y_p$, denote an increasing subsequence of the non-degenerated classes of the

[1] The inequalities are different in the case of $y \in \{1, Y\}$, however, the analysis remains similar thus it is omited here.

[2] We use convention that a sum is zero if its upper index is less than the lower one.

MORD classifier $h'(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{b})$. For arbitrary $\boldsymbol{x}_{y_i} \in \mathcal{X}_{y_i} = \{\boldsymbol{x} \in \mathbb{R}^n \mid h'(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{b}) = y_i\},\ i = 1, \ldots, p$, it holds that

$$\begin{aligned} \langle \boldsymbol{w}, \boldsymbol{x}_{y_i} \rangle y_i + b_{y_i} &> \langle \boldsymbol{w}, \boldsymbol{x}_{y_{i-1}} \rangle y_{i-1} + b_{y_{i-1}}, \\ \langle \boldsymbol{w}, \boldsymbol{x}_{y_i} \rangle y_i + b_{y_i} &\geq \langle \boldsymbol{w}, \boldsymbol{x}_{y_{i+1}} \rangle y_{i-1} + b_{y_{i+1}}, \end{aligned} \tag{25}$$

It follows that

$$\tfrac{b_{y_i} - b_{y_{i+1}}}{y_{i+1} - y_i} \geq \langle \boldsymbol{w}, \boldsymbol{x}_{y_i} \rangle > \tfrac{b_{y_{i-1}} - b_{y_i}}{y_i - y_{i-1}}, \quad i = 1, \ldots, p-1.$$

Thus, for any MORD classifier $h'(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{b})$ with non-degenerated classes $y_1, \ldots, y_p$, it holds that

$$\tfrac{b_{y_{p-1}} - b_{y_p}}{y_p - y_{p-1}} > \cdots > \tfrac{b_{y_{i-1}} - b_{y_i}}{y_i - y_{i-1}} > \cdots > \tfrac{b_{y_1} - b_{y_2}}{y_2 - y_1}. \tag{26}$$

We are now ready to proof the second part of the theorem stating that for any $\boldsymbol{w} \in \mathbb{R}^n$, $\boldsymbol{b} \in \mathbb{R}^Y$ and the admissible vector $\boldsymbol{\theta} \in \Theta$ computed by the formula (7) the equality $h(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{\theta}) = h'(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{b})$ holds $\forall \boldsymbol{x} \in \mathbb{R}^n$. It is enough to show that for arbitrary $\boldsymbol{x} \in \mathcal{X}$ the ORD classifier $h(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{\theta})$ outups y_i iff the MORD classifier $h'(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{b})$ outputs the same output y_i.

First, suppose the MORD classifier $h'(\boldsymbol{x}; \boldsymbol{w}, \boldsymbol{b})$ outputs $y_i \in \mathcal{Y}$ for some $\boldsymbol{x} \in \mathcal{X}$. We want to show that the ORD classifier $h(\boldsymbol{x}; \boldsymbol{w}, \boldsymbol{\theta})$ outputs the same label y_i. We shall analyse only the cases $1 < i < p$, however, the prove for $i \in \{1, p\}$ is similar and hence omitted. The equality $h'(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{b}) = y_i$ implies that

$$\begin{aligned} \langle \boldsymbol{w}, \boldsymbol{x} \rangle y_i + b_{y_i} &> \langle \boldsymbol{w}, \boldsymbol{x} \rangle y_{i-1} + b_{y_{i-1}}, \\ \langle \boldsymbol{w}, \boldsymbol{x} \rangle y_i + b_{y_i} &\geq \langle \boldsymbol{w}, \boldsymbol{x} \rangle y_{i+1} + b_{y_{i+1}}, \end{aligned} \tag{27}$$

which is equivalent to $\frac{b_{y_i} - b_{y_{i+1}}}{y_{i+1} - y_i} \geq \langle \boldsymbol{w}, \boldsymbol{x} \rangle > \frac{b_{y_{i-1}} - b_{y_i}}{y_i - y_{i-1}}$ and after combining with (7) we see that the ORD classifier $h(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{\theta})$ outputs the same y_i.

Second, suppose the ORD classifier $h(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{\theta})$ outputs y_i for some arbitrary $\boldsymbol{x} \in \mathcal{X}$, i.e. $\frac{b_{y_i} - b_{y_{i+1}}}{y_{i+1} - y_i} \geq \langle \boldsymbol{w}, \boldsymbol{x} \rangle > \frac{b_{y_{i-1}} - b_{y_i}}{y_i - y_{i-1}}$ holds. To show that MORD classifier $h'(\boldsymbol{x}; \boldsymbol{w}, \boldsymbol{\theta})$ outputs the same y_i it is enough to prove that the system

$$\langle \boldsymbol{w}, \boldsymbol{x} \rangle y_i + b_{y_i} > \langle \boldsymbol{w}, \boldsymbol{x} \rangle y_j + b_{y_j},\ \forall y_j < y_i, \tag{28}$$

$$\langle \boldsymbol{w}, \boldsymbol{x} \rangle y_i + b_{y_i} \geq \langle \boldsymbol{w}, \boldsymbol{x} \rangle y_j + b_{y_j},\ \forall y_j > y_i \tag{29}$$

holds. Indeed, from inequality $\langle \boldsymbol{w}, \boldsymbol{x} \rangle > \frac{b_{y_{i-1}} - b_{y_i}}{y_i - y_{i-1}}$ after some algebra and applying (26) (after third line) we have

$$\begin{aligned} \langle \boldsymbol{w}, \boldsymbol{x} \rangle (y_i - y_j) &> (y_i - y_j) \tfrac{b_{y_{i-1}} - b_{y_i}}{y_i - y_{i-1}} \\ &= (-y_j + y_{j+1} - y_{j+1} + \cdots + y_{i-1} - y_{i-1} + y_i) \tfrac{b_{y_{i-1}} - b_{y_i}}{y_i - y_{i-1}} \\ &= (y_{j+1} - y_j) \tfrac{b_{y_{i-1}} - b_{y_i}}{y_i - y_{i-1}} + \cdots + (y_i - y_{i-1}) \tfrac{b_{y_{i-1}} - b_{y_i}}{y_i - y_{i-1}} \\ &> (y_{j+1} - y_j) \tfrac{b_{y_j} - b_{y_{j+1}}}{y_{j+1} - y_j} + \cdots + (y_i - y_{i-1}) \tfrac{b_{y_{i-1}} - b_{y_i}}{y_i - y_{i-1}} \\ &= b_{y_j} - b_{y_{j+1}} + b_{y_{j+1}} - \cdots - b_{y_{i-1}} + b_{y_{i-1}} - b_{y_i} = b_{y_j} - b_{y_i}\ , \end{aligned}$$

from which the inequalities (28) follow for $\forall y_j < y_i$. The proof of the inequalities (29) is analogical. ■